

AMENDMENTS TO THE CLAIMS:

Please amend claims 1, 21 and 37 and cancel claims 4, 24 and 40 as set forth in the following listing of claims. This listing of claims will replace all prior versions and listings of claims in the application.

1. (Currently amended) A computer-implemented method for determining a relationship between objects related to a common information model, the objects including at least a first and second instance and an association that represents an instance of an association class, the method comprising:

creating, for the first instance, a reverse link that defines a relationship between the first instance and the association, wherein the instance is associated with a first wrapper defining the reverse link, wherein creating the reverse link includes:

defining a pointer in a first table of the first wrapper that references a second table; and

defining a pointer in the second table that references the instance of the association class; and

determining a relationship between the first and second instances based on the reverse link.

2. (Original) The method of claim 1, wherein each association reflects a relationship between a respective association and a corresponding associated object.

3-4. (Canceled).

5. (Currently amended) The method of claim [[4]] 1, wherein determining a relationship includes:

collecting a reference reflecting a relationship between the association and the second instance based on the pointer in the second table.

6. (Previously presented) A method for maintaining reverse links in a object-oriented environment including class instances and associations, the method comprising:

for each class instance associated with N instances of an association class that each reference the class instance, wherein N represents an integer value greater than or equal to one:

(i) creating a first level wrapper table including a pointer to a second level wrapper table associated with the association class; and

(ii) creating N pointers in the second level wrapper table that each reference an individual instance of the association class.

7. (Previously presented) The method of claim 6, further comprising:

for each new class instance and corresponding new association class instance that reference the new class instance that is created:

(iii) creating a new first wrapper level table including a pointer to a new second level wrapper table associated with the association class; and

(iv) creating a pointer in the new second level wrapper table that references the new instance of the association class.

8. (Previously presented) The method of claim 6, further comprising:

for each new class instance and corresponding new association class instance that is created:

(iii) determining all instances of the association class that reference the new class instance;

(iv) creating a new first level wrapper table including a pointer to a new second level wrapper table associated with the association class; and

(v) creating a pointer in the new second level wrapper table for each instance of the association class determined in step (iii).

9. (Previously presented) A method for maintaining reverse links in an object-oriented environment including instances, association classes and association class instances, the method comprising:

for each class instance associated with N association class classes that each includes include at least one association class instance that references the class instance, wherein N represents an integer value greater than or equal to one:

creating a first level wrapper table including N pointers to N second level wrapper tables associated with the association classes.

10. (Previously presented) A method for maintaining reverse links in a object-oriented environment including class instances and associations, the method comprising:

for each class instance associated with N instances of a first association class that each reference the class instance, and X instances of a second association class that each reference the class instance, wherein N and X represent integer values greater than or equal to one:

(i) creating a first level wrapper table including:

a first pointer to a second level wrapper table associated with the first association class, and

a second pointer to a second level wrapper table associated with the second association class;

(ii) creating N pointers, in the second table wrapper table associated with the first association class, that each reference an individual instance of the first association class; and

(iii) creating X pointers, in the second table level wrapper table associated with the second association class, that each reference an individual instance of the second association class.

11. (Previously presented) A method for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

receiving an association traversal request for a class instance; and
performing an association traversal process based on pointer information reflecting a relationship between the class instance and all association instances that reference the class instance, wherein the association traversal process includes:
accessing a first table including a pointer to a second table; and
accessing the second table, using the pointer, to obtain pointers to each association instance that references the class instance.

12. (Canceled).

13. (Previously presented) The method of claim [[12]] 11, wherein the association traversal process further includes:

for each association instance pointed to by the second table:
collecting a reference to another class instance that is referenced by the association instance.

14. (Previously presented) A method for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

receiving an association traversal request for a class instance;

obtaining pointers to each association instance that references the class instance from a wrapper table corresponding to the class instance; and

collecting, from the wrapper table, references to other class instances that are referenced by the association instance.

15. (Previously presented) A method for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

receiving an association traversal request for a class instance; and

obtaining pointers to each association instance that references the class instance from a wrapper table corresponding to the class instance.

16. (Previously presented) In a system comprising a client and a server, a method for performing association traversals performed by the client, comprising:
generating a request for relationship information associated with a selected class instance; and

receiving a response including information reflecting a relationship between the selected class instance and other class instances that are referenced by the same association class instances, wherein the response was is generated using pointers defined in a table associated with the selected class instance that reference the common association class instances.

17. (Original) In a system comprising a client and a server, a method for performing association traversals performed by the server, comprising;

receiving a request for relationship information associated with a selected class instance; and

generating a response including information reflecting a relationship between the selected class instance and other class instances that are referenced by the same association class instances, based on pointers defined in a table associated with the selected class instance that reference the common association class instances.

18. (Original) A system for performing association traversals, comprising:
a client for generating a request for information reflecting the relationship
between selected objects defined in a repository; and
a server including:
the repository for storing objects, wherein a set of the objects are
associated with object wrappers, and
an object manager for processing the request by using an object wrapper
associated with one object of the selected objects.

19. (Original) The system of claim 18, wherein the object wrapper associated
with one object includes pointers to all association instances that reference the one
object.

20. (Original) The system of claim 19, wherein the object manager uses the
pointers to collect references to the selected objects.

21. (Currently amended) A system for traversing associations in a common information model implemented environment, the model comprising at least a first and second instance and an association that represents an instance of an association class, comprising:

means for creating, for the first instance, a reverse link that defines a relationship between the first instance and the association, wherein the instance is associated with a first wrapper defining the reverse link, wherein the means for creating the reverse link includes:

means for defining a pointer in a first table of the first wrapper that references a second table; and

means for defining a pointer in the second table that references the instance of the association class; and

means for determining a relationship between the first and second instances based on the reverse link.

22. (Original) The system of claim 21, wherein each association reflects a relationship between a respective association and a corresponding associated object.

23-24. (Canceled).

25. (Currently amended) The system of claim ~~[[24]]~~ 21, wherein the means for determining a relationship includes:

means for collecting a reference reflecting a relationship between the association and the second instance based on the pointer in the second table.

26. (Previously presented) A system for maintaining reverse links in a object-oriented environment including class instances and associations, comprising:

means for creating, for each class instance associated with N instances of an association class that each reference the class instance, wherein N represents an integer value greater than or equal to one, a first level wrapper table including a pointer to a second level wrapper table associated with the association class; and

means for creating N pointers in the second level wrapper table that each point to an individual instance of the association class.

27. (Previously presented) The system of claim 26, further comprising:

means for creating, for each new class instance and corresponding new association class instance that reference the new class instance that is created, a new first level wrapper table including a pointer to a new second level wrapper table associated with the association class; and

means for creating a pointer in the new second level wrapper table that references the new instance of the association class.

28. (Previously presented) The system of claim 26, further comprising:

means for determining, for each new class instance and corresponding new association class instance that is created, all instances of the association class that reference the new class instance;

means for creating a new first level wrapper table including a pointer to a new second level wrapper table associated with the association class; and

means for creating a pointer in the new second level wrapper table for each instance of the association class determined by the means for determining.

29. (Previously presented) A system for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

means for receiving an association traversal request for a class instance; and

means for performing an association traversal process based on pointer information reflecting a relationship between the class instance and all association instances that reference the class instance, wherein the means for performing an association traversal process includes:

means for accessing a first table including a pointer to a second table; and

means for accessing the second table, using the pointer, to obtain pointers to each association instance that references the class instance.

30. (Canceled).

31. (Previously presented) The system of claim [[30]] 29, wherein the means for performing an association traversal process further includes:

means for collecting, for each association instance pointed to by the second table, a reference to another class instance that is referenced by the association instance.

32. (Previously presented) A system for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

means for receiving an association traversal request for a class instance;

means for obtaining pointers to each association instance that references the class instance from a wrapper table corresponding to the class instance; and

means for collecting, from the wrapper table, references to other class instances that are referenced by the association instance.

33. (Previously presented) A system for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

means for receiving an association traversal request for a class instance; and

means for obtaining pointers to each association instance that references the class instance from a wrapper table corresponding to the class instance.

34-36. (Canceled).

37. (Currently amended) A computer-readable medium including instructions for performing a method, when executed by a processor, for determining a relationship between objects related to a common information model, the objects including at least a first and second instance and an association that represents an instance of an association class, the method comprising:

creating, for the first instance, a reverse link that defines a relationship between the first instance and the association, wherein the instance is associated with a first wrapper defining the reverse link, and wherein creating the reverse link includes:

defining a pointer in a first table of the first wrapper that references a second table; and

defining a pointer in the second table that references the instance of the association class; and

determining a relationship between the first and second instances based on the reverse link.

38. (Original) The computer-readable medium of claim 37, wherein each association reflects a relationship between a respective association and a corresponding associated object.

39-40. (Canceled).

41. (Currently amended) The computer-readable medium of claim 37, wherein the step of determining a relationship includes:

collecting a reference reflecting a relationship between the association and the second instance based on the pointer in the second table.

42. (Previously presented) A computer-readable medium including instructions for performing a method, when executed by a processor, for maintaining reverse links in a object-oriented environment including class instances and associations, the method comprising:

for each class instance associated with N instances of an association class that each reference the class instance, wherein N represents an integer value greater than or equal to one:

(i) creating a first level wrapper table including a pointer to a second level wrapper table associated with the association class; and

(ii) creating N pointers in the second level wrapper table that each reference an individual instance of the association class.

43. (Previously presented) The computer-readable medium of claim 42, wherein the method further comprises:

for each new class instance and corresponding new association class instance that reference the new class instance that is created:

(iii) creating a new first level wrapper table including a pointer to a new second level wrapper table associated with the association class; and

(iv) creating a pointer in the new second level wrapper table that references the new instance of the association class.

44. (Previously presented) The computer-readable medium of claim 42, wherein the method further comprises:

for each new class instance and corresponding new association class instance that is created:

(iii) determining all instances of the association class that reference the new class instance;

(iv) creating a new first level wrapper table including a pointer to a new second level wrapper table associated with the association class; and

(v) creating a pointer in the new second level wrapper table for each instance of the association class determined in step (iii).

45. (Previously presented) A computer-readable medium including instructions for performing a method, when executed by a processor, for maintaining reverse links in a object-oriented environment including class instances and associations, the method comprising:

for each class instance associated with N association classes that each includes include at least one association class instance that references the class instance, wherein N represents an integer value greater than or equal to one:

creating a first level wrapper table including N pointers to N second level wrapper tables associated with the association classes.

46. (Previously presented) A computer-readable medium including instructions for performing a method, when executed by a processor, for maintaining reverse links in a object-oriented environment including class instances and associations, the method comprising:

for each class instance associated with N instances of a first association class that each reference the class instance, and X instances of a second association class that each reference the class instance, wherein N and X represent integer values greater than or equal to one:

- (i) creating a first level wrapper table including:
 - a first pointer to a second level wrapper table associated with the first association class, and
 - a second pointer to a second level wrapper table associated with the second association class;
- (ii) creating N pointers, in the second table wrapper table associated with the first association class, that each reference an individual instance of the first association class; and
- (iii) creating X pointers, in the second table level wrapper table associated with the second association class, that each reference an individual instance of the second association class.

47. (Previously presented) A computer-readable medium including instructions for performing a method, when executed by a processor, for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

- receiving an association traversal request for a class instance; and
- performing an association traversal process based on pointer information reflecting a relationship between the class instance and all association instances that reference the class instance, wherein the association traversal process includes:

- accessing a first table including a pointer to a second table; and
 - accessing the second table, using the pointer, to obtain pointers to each association instance that references the class instance.

48. (Canceled).

49. (Previously presented) The computer-readable medium of claim [[48]] 47, wherein the association traversal process further includes:

- for each association instance pointed to by the second table:
 - collecting a reference to another class instance that is referenced by the association instance.

50. (Previously presented) A computer-readable medium including instructions for performing a method, when executed by a processor, for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

receiving an association traversal request for a class instance;

obtaining pointers to each association instance that references the class instance from a wrapper table corresponding to the class instance; and

collecting, from the wrapper table, references to other class instances that are referenced by the association instance.

51. (Previously presented) A computer-readable medium including instructions for performing a method, when executed by a processor, for performing association traversals in an object-oriented environment including a plurality of class instances and association instances, comprising:

receiving an association traversal request for a class instance; and

obtaining pointers to each association instance that references the class instance from a wrapper table corresponding to the class instance.

52. (Previously presented) In a system comprising a client and a server, a computer-readable medium including instructions for performing a method, when executed by a processor, for performing association traversals performed by the client, comprising:

generating a request for relationship information associated with a selected class instance;

receiving a response including information reflecting a relationship between the selected class instance and other class instances that are referenced by the same association class instances, wherein the response was is generated using pointers defined in a table associated with the selected class instance that reference the common association class instances.

53. (Original) In a system comprising a client and a server, a computer-readable medium including instructions for performing a method, when executed by a processor, for performing association traversals performed by the server, comprising;

receiving a request for relationship information associated with a selected class instance;

generating a response including information reflecting a relationship between the selected class instance and other class instances that are referenced by the same association class instances, based on pointers defined in a table associated with the selected class instance that reference the common association class instances.

54-56. (Canceled).